# **RCNN**

RCNN(Region with CNN feature)

论文：Rich feature hierarchies for accurate object detection and semantic segmentation

RCNN可以说是深度学习目标检测的开山之作。作者Ross Girshick多次在Pascal voc目标检测竞赛中夺冠。

2014年前，传统的目标检测算法准确率在30左右，RCNN之后最多可以提升30左右的准确率。

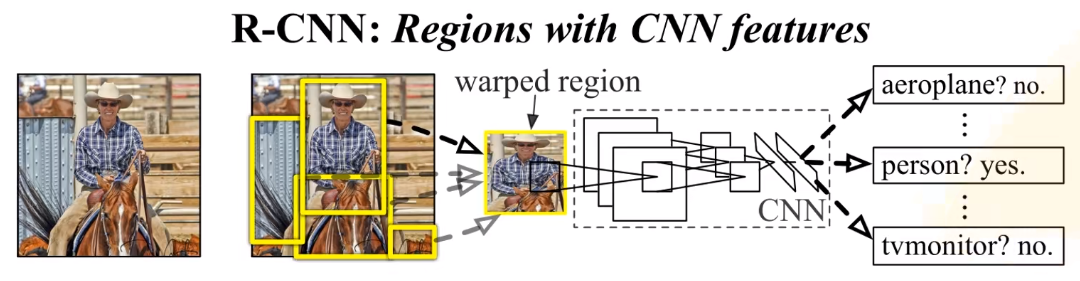
RCNN算法流程可以分为4个步骤：

1.一张图像生成1k~2k个候选区域(使用Selective Search方法)

2.对每个候选区域使用深度网络提取特征

3.特征送入每一类的SVM分类器，判别是否属于该类

4.使用回归器精细修正候选框位置(通过SS算法框出来的没那么准，用回归器修正候选框位置)



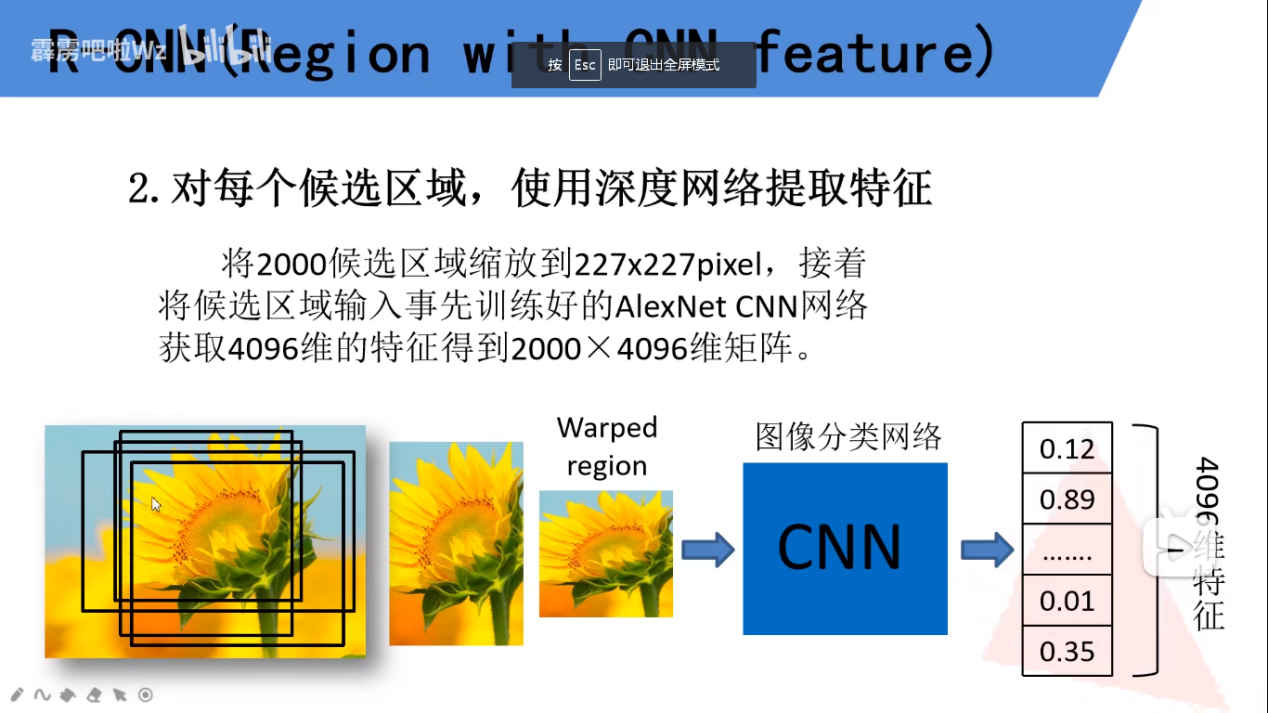
**1.候选区域的生成**

利用SS算法分割得到一些原始区域，然后通过一些合并策略将这些区域合并。得到的区域可能包含需要的物体



**2.对每个候选区域，使用深度网络提取特征**

通过SS方法可以得到约2000个候选区域，并将2000个候选区域resize成227\*227。再放到训练好的AlexNet CNN中，获得4096维的特征向量，2000个图片就是2000个\*4096维特征矩阵。

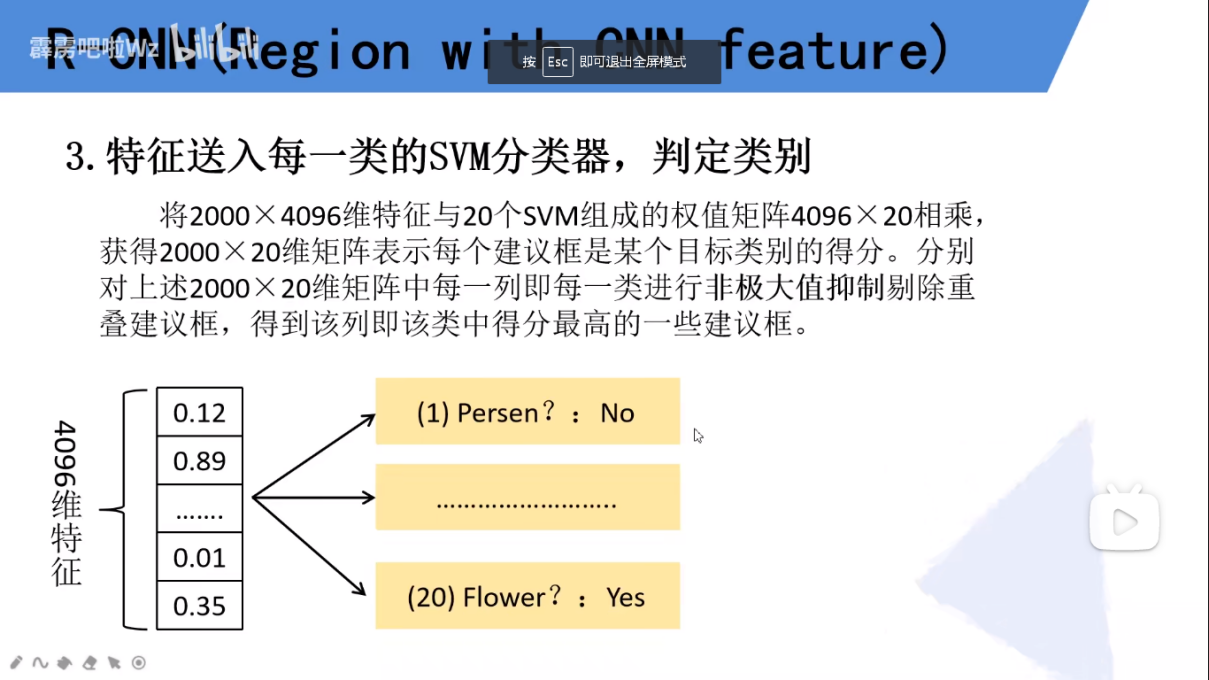


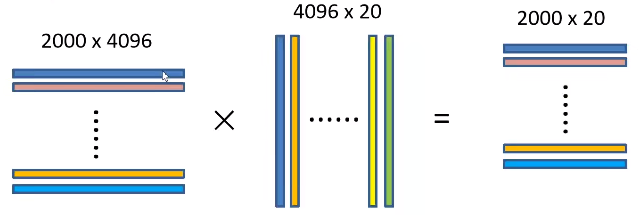
3.特征送入每一类的SVM分类器，判定类别

SVM是一个二分类的分类器，将全连接层替换为SVM分类器，SVM分类器是事先专门训练好的。

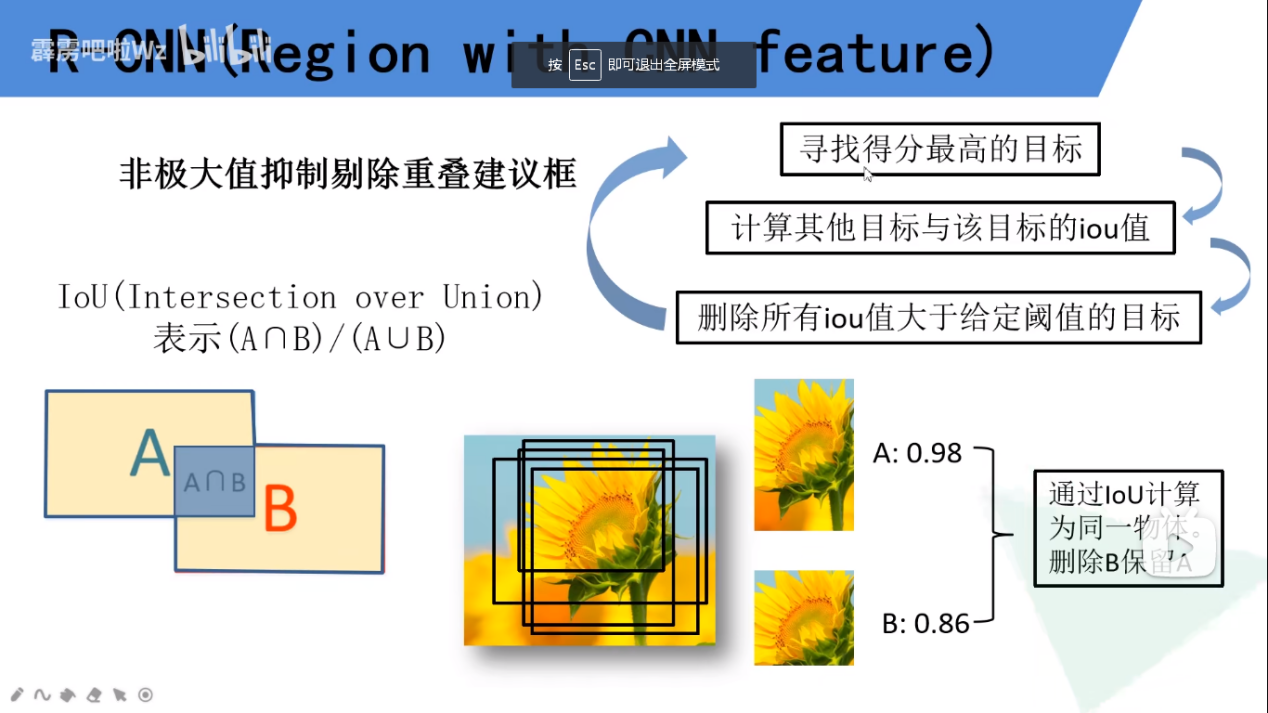
将2000\*4096(特征矩阵)的特征矩阵与20个SVM(权值矩阵)组成的权值矩阵4096\*20相乘。获得2000\*20(评分矩阵)特征矩阵(每一个图片都会有20个SVM分类，根据分数判定是否属于某一类)表示每个建议框是某个目标的得分(框是某类的得分)。对上述2000\*20矩阵中每一类进行非极大抑制剔除重叠的建议框，得到该列即该类中纷纷最高的建议框。

2000(行)个\*20维的特征矩阵，每行特征矩阵对应的是一个图片的特征，20维代表分成20类，里面的数值是得分，每个对应一个SVM，一共20类对应20个SVM，SVN二分类来判定是否是当前这一类





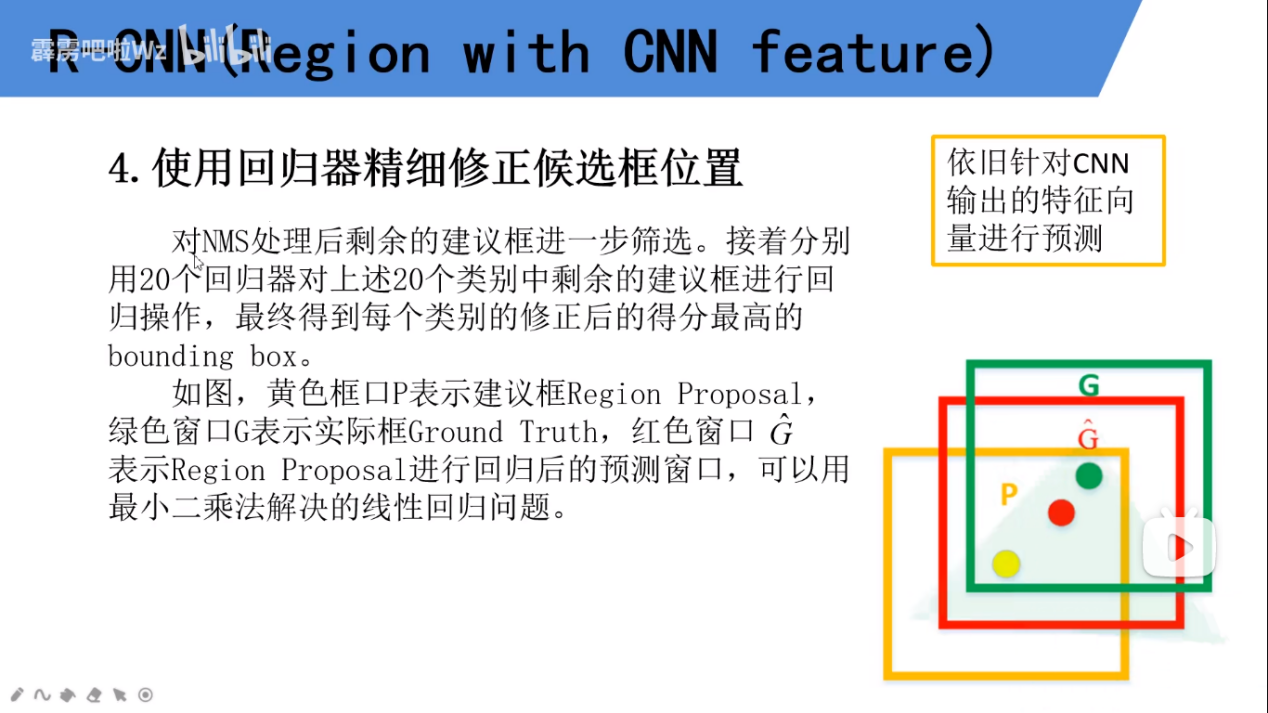
A是候选框中评分最高的一个，B是随便一个？计算A和B的IoU，如果大于某个阈值，则认为这两个目标是同一个目标，则保留评分最高的A删B。最后获得最完美的一个边界框

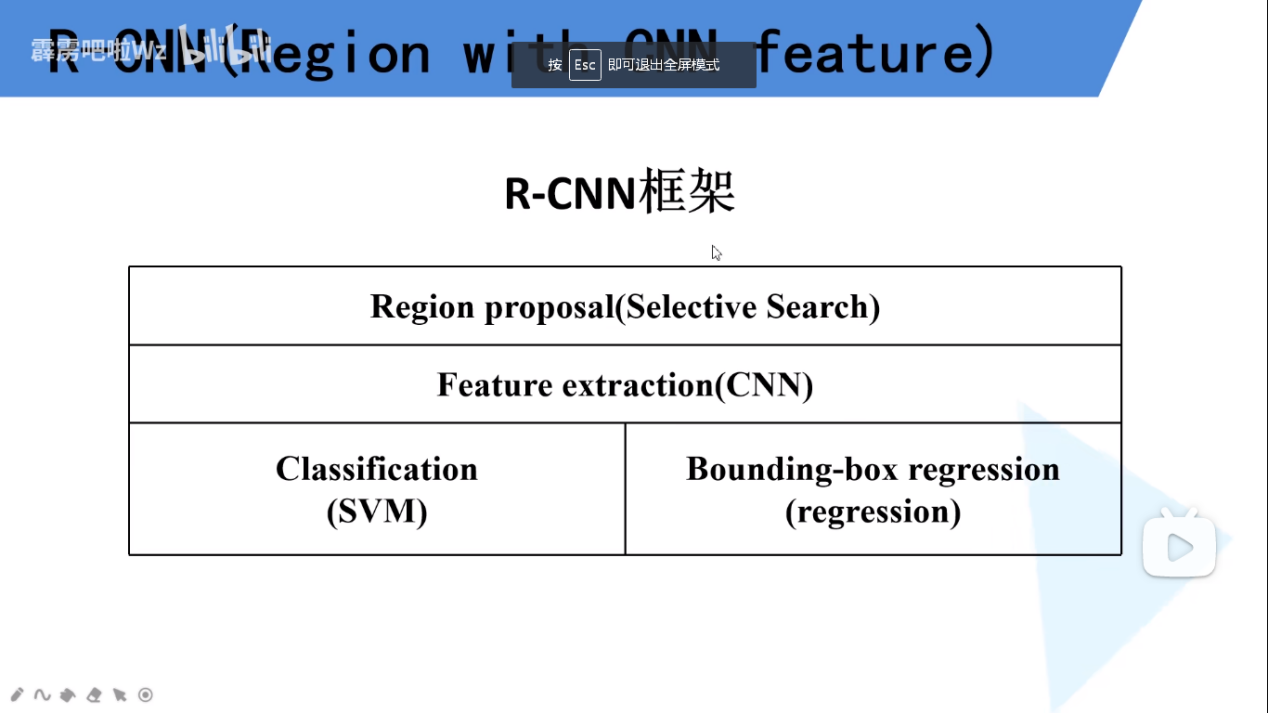


**4.使用回归器精细修正候选框位置**

边界框回归器也是提前训练好的

黄色的是建议框(region proposal)，绿色的是实际框(ground truth),红色的是回归后的预测窗口





**RCNN问题：**

1. 测试慢：一张图在多核cpu要53秒(对于现在来说已经很慢了，但是在原论文那个时代，好像挺快)。提取候选框就要2s。SS算法会找出大量的重叠区域(比如中间花芯的位置)，在卷积时会产生部分重复的特征向量。

2. 训练慢：过程及其繁琐

3. 训练所需空间大



**bounding box regression边框回归：**

<https://blog.csdn.net/zijin0802034/article/details/77685438/>

ground truth:绿色的框，框起来的是整个物体

region proposal:用SS算法框出来的框，与ground truth不同，region proposal框的来的框小很多，更能代表这个物体

bounding box regression:基于ground truth，将region proposal框进行回归预测，框出来的更准